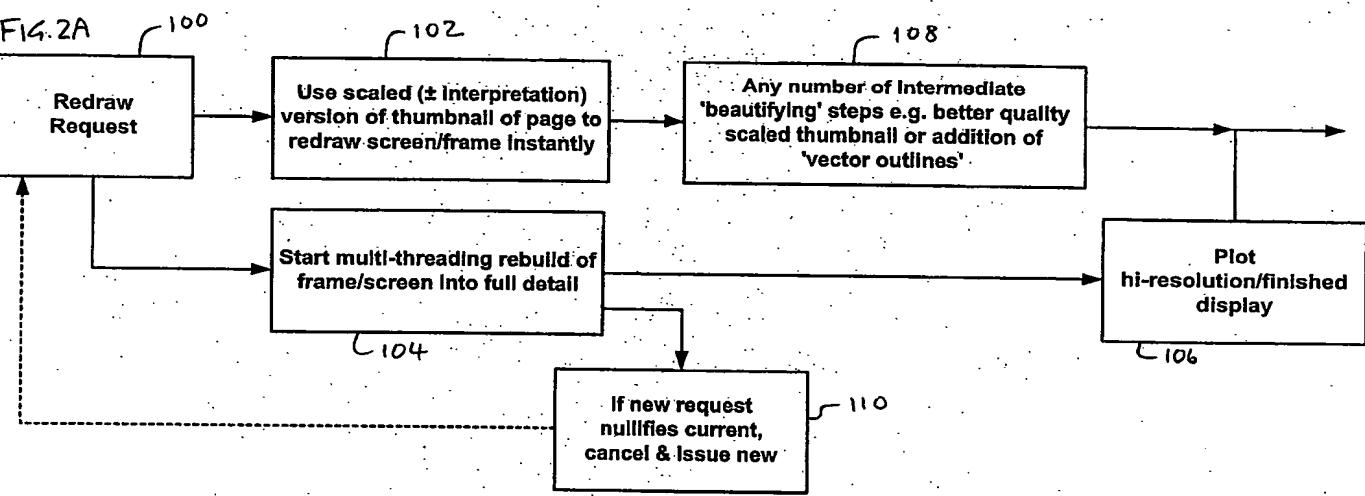
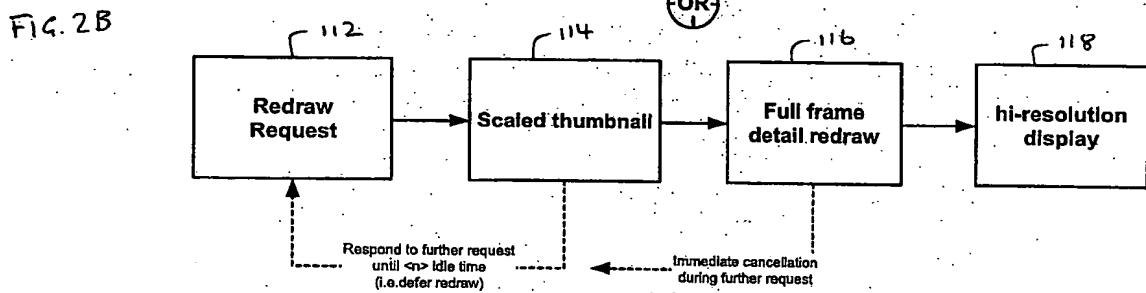


FIG. 1

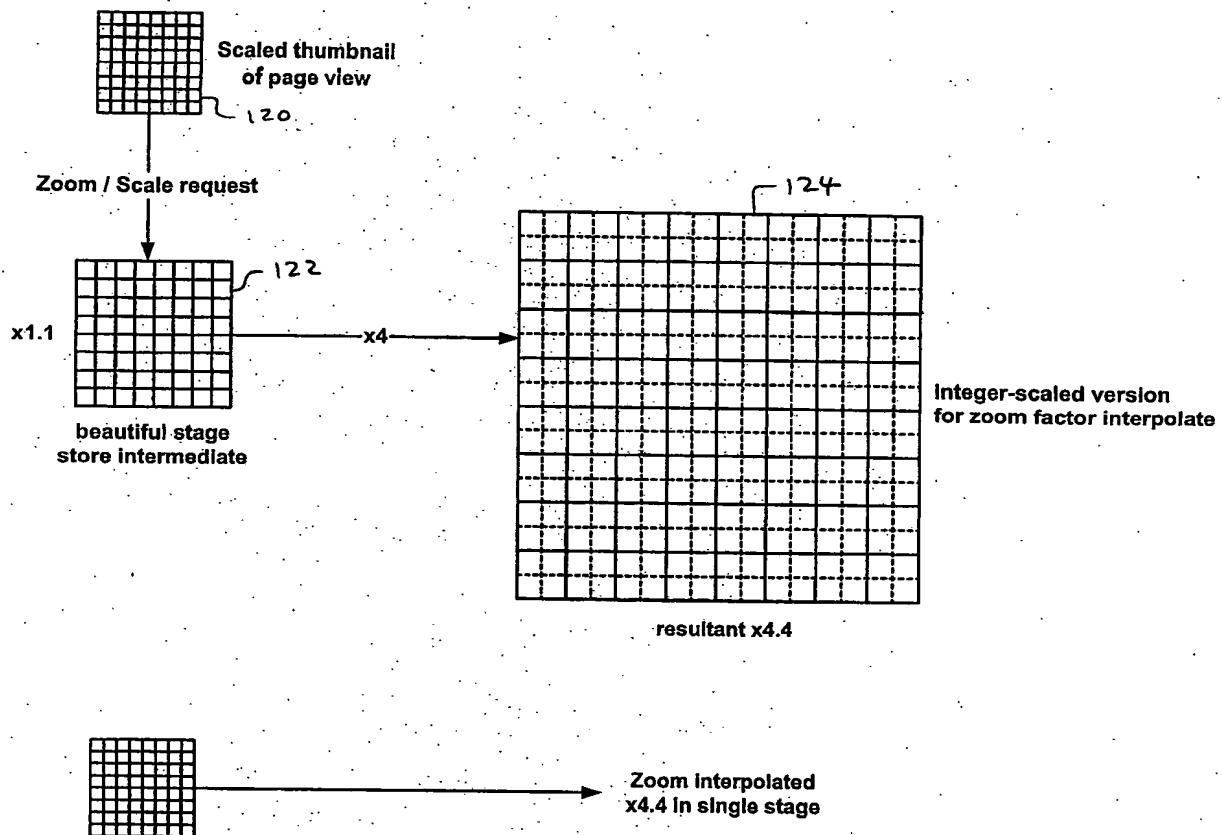
0093354933 - 0414601



2/8

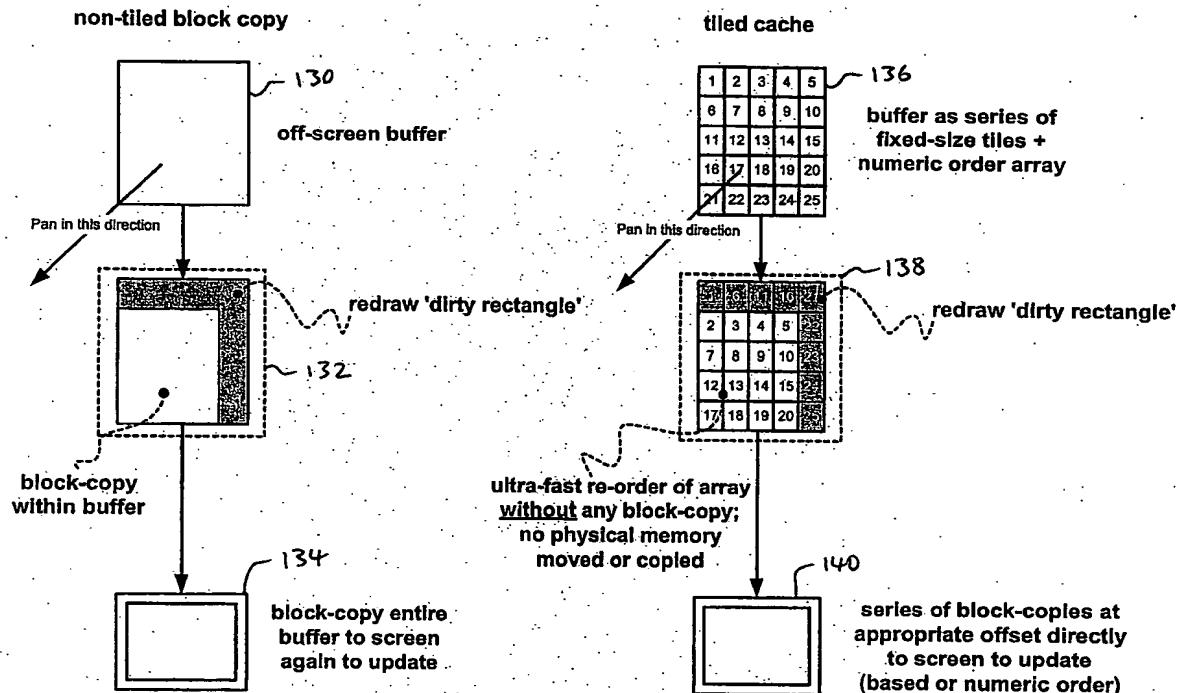


0141071401



- Intermediate stage 'infrequent' & therefore can use beautiful/detailed scaling, versus rapid/cruder final or single stage scale

FIG. 4A



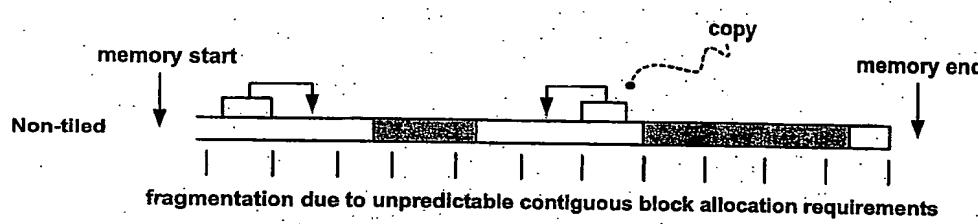


FIG. 5A

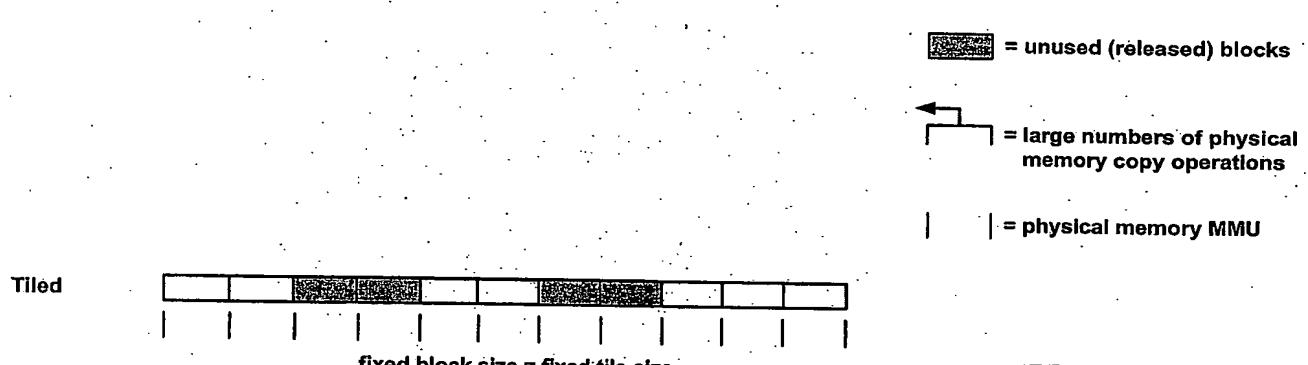
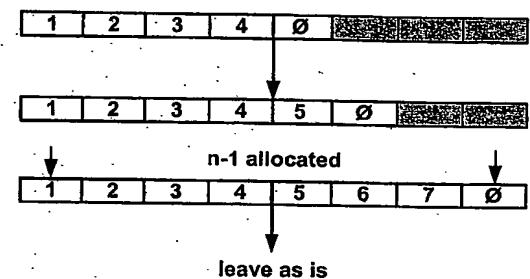
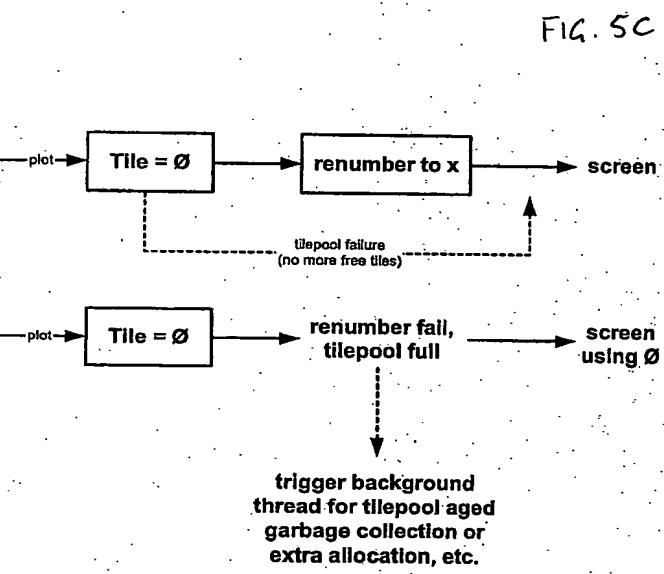


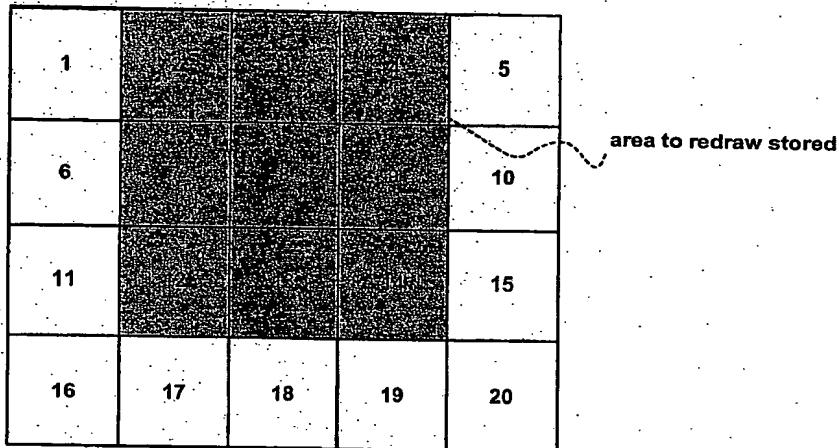
FIG. 5B

unlimited amount of fragmentation has no effect on usability
no copy operations required (for buffer re-centering Fig.24)
potential perfect synchronisation with MMU predictability
& extendibility of pool (!)



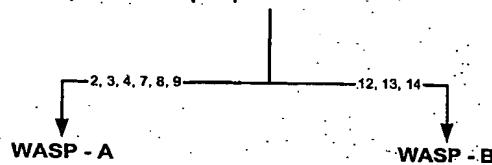
0 9 8 3 5 4 8 3 0 4 1 0 5 0

FIG. 6

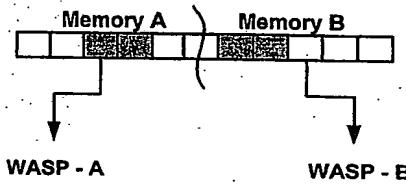


7/8
tiles 1-10 handled by WASP-A
tiles 11-20 handled by WASP-B

redraw request → pre-processor



based on memory map of Fig. 5B
split pool accordingly regardless of
screen position



00101101010101000000000000000000

FIG. 7

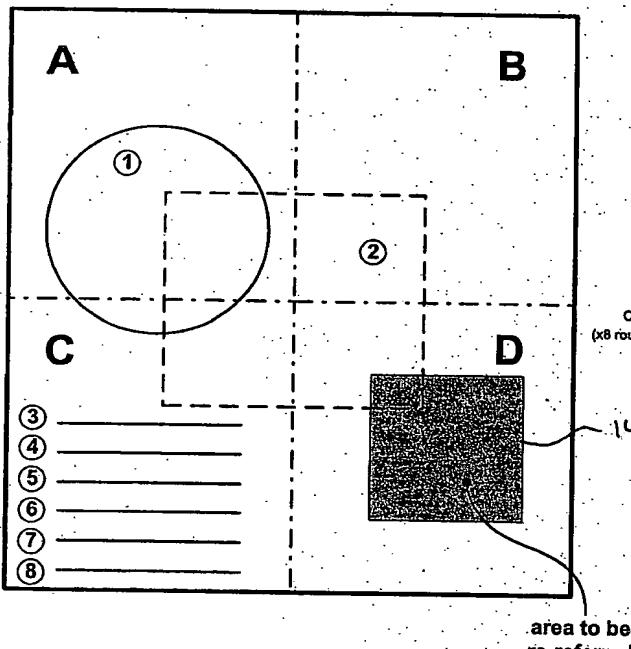


FIG. 8

Without Zoning:

142 to update left diagram.

Extract Item ① from list

Check bounding box if fits within

Yes

Intersect Bounding Box & plot object

Continue next item
(x8 round loop in this example)

No

8/8

With Zoning:

Intersect with zones

Concatenate item from zone(s) list, here = D, get ②

Vastly reduces in many cases amount of objects extracted & compared.

Ratio of zone size & typical object size is critical.

Check bounding box if fits

As above

(x1 loop in this example)

Common case of many small locally clustered objects (text / gradfills) good example of beneficiaries.